EXHIBIT 1

A Constrained Multipath Traffic Engineering Scheme for MPLS Networks

Youngseok Lee, Yongho Seok, Yanghee Choi
School of Computer Science and Engineering
Seoul National University
San 56-1, Shillim-dong, Kwanak-gu
Seoul, Korea, 151-742

Changhoon Kim
Dept. of Internet Technologies
Electronic Telecommunication Research Institute
161, Gajeong-dong, Yusong-gu
Taejon, Korea, 305-350

Abstract— **A traffic engineering problem consists of setting up paths between the edge nodes of the network to meet traffic demands while optimizing the network performance. It is known that total traffic throughput in a network, hence the resource utilization, can be maximized if the traffic demand is split over multiple paths. However, the problem formulation and practical algorithms, which calculate the paths and the traffic split ratio taking the route constraints or policies into consideration, have not been much touched. This paper proposes practical algorithms that find near optimal paths satisfying the given traffic demand under constraints such as maximum hop count, and preferred or not-preferred node/link list. The mixed integer programming formulation also calculates the traffic split ratio for the multiple paths. The problems are solved with the split ratio of continuous or discrete values. However, the split ratio solved with discrete values (0.1, 0.2 etc.) are more suitable for easy implementation at the network nodes. The proposed algorithms are applied to the multi-protocol label switching (MPLS) that permits explicit path setup. The paths and split ratio are calculated off-line, and passed to MPLS edge routers for explicit label-switched path (LSP) setup. The proposed schemes are tested in a large-scale fictitious backbone network. The experiment results show that the proposed algorithms are fast and superior to the conventional shortest path algorithm in terms of maximum link utilization, total traffic volume, and number of required LSPs.**

Index Terms— **MPLS, Traffic Engineering, Multipath, Load Balancing**

I. Introduction

A traffic engineering problem in the Internet consists of setting up paths between the edge routers in a network to meet traffic demands while achieving low congestion and optimizing the utilization of network resources. In practice, the usual key objective of traffic engineering is to minimize the utilization of the most heavily used link in the network, or the maximum of link utilization (denoted as α). As the maximum link utilization qualitatively expresses that congestion sets in when link utilization increases higher, it is important to minimize the link utilization throughout the network so that no bottleneck link exists. It is known that this problem of minimizing the maximum link utilization can be solved by the multi-commodity network flow formulation of optimal routing, which leads to splitting traffic over multiple paths between source-destination pairs [1].

This work was supported in part by the Brain Korea 21 project of Ministry of Education, in part by the National Research Laboratory project of Ministry of Science and Technology, and in part by Electronic Telecommunication Research Institute, 2001, Korea.

Multipath routing provides increased bandwidth, and the network resources are more efficiently used than in the case of the single shortest path algorithm. Multipath routing has been incorporated in recently developed or proposed routing protocols. The easiest extension to multipath routing is to use the equal-cost multiple shortest paths when calculating the shortest path, which is known as Equal-Cost Multi-Path (ECMP) [4] routing. This is explicitly supported by several routing protocols such as Open Shortest Path First (OSPF) [4] and Intermediate System to Intermediate System (IS-IS) [5]. Some router implementations allow equal-cost multipath with Routing Information Protocol (RIP) and other routing protocols [6]. In MPLS networks [2], multiple paths can be used to forward packets belonging to the same "forwarding equivalent class (FEC)" by explicit routing.

However, the problem formulation and practical algorithms, which calculate the paths and the traffic split ratio taking the route constraints or policies into consideration, have not been much touched. Specifically, multipath routing algorithms considering constraints such as discrete split ratio, maximum hop count, and link/node affinity have not much studied.

This paper proposes practical algorithms that find near optimal paths satisfying given traffic demands. Traffic demands are expressed in the matrix form where entry (i, j) represents the average traffic volume from edge router i to j, in bps. For Virtual Private Network (VPN) application, the entry may be the requested amount of bandwidth reservation. Traffic split ratios for the calculated paths are also obtained from the proposed algorithms. The split ratio is fed to the routers for dividing the traffic of the same source-destination pair to the multiple paths. For easy implementation at the routers, we suggest that multipath routers use only discrete values for the split ratio. The split ratio is usually a discrete value with coarse granularity, because partitioning a traffic demand can be done by adjusting the output range of the hashing function of dynamically changing IP flows [7]. Moreover, in the multi-protocol lambda switching (MPλS) network, an optical cross-connect (OXC) can support only a relatively small number of optical channel (OCh) trails which will have discrete bandwidth granularities (e.g., OC-12, OC-48, OC-192, and OC-768).

When calculating the paths and the split ratio, we also consider constraints such as the maximum hop count allowed for LSPs and the preferred or not-preferred node or link list. The maximum hop count constraint is for reducing the delay and total traffic volume. On the other hand, the node or link affinity

constraint, such as including/excluding preferred/not-preferred nodes/links when calculating paths between a node pair, is valuable in reflecting routing policies. The node/link affinity constraint is a subset of the resource class affinity in MPLS networks [3]. When nodes or links have resource class attributes or colors, the network administrator may enforce LSPs for a traffic demand to be established by including or excluding specific colored links or nodes.

The mixed integer programming (MIP) formulation of the multiple-path finding problem that we propose in this paper carries out the computation in three steps. First, the traffic demand is split over the network links so that the maximum link utilization in the network is minimized while the traffic demand is satisfied and other constraints are observed. Although traffic bifurcation is allowed, the total network resources necessary for multipath routing should be minimized in order to accept more connection requests in the future. Thus, the second step consists of finding a rearranged traffic split pattern under the constraint of α obtained in the first round, that minimizes the total resources used for assigning all the traffic demands. After two steps, finding paths from the split pattern is easily performed by using the algorithm for maximum flow path for each traffic demand. We apply the proposed algorithms for the MPLS networks where multiple LSPs can be explicitly established by a signaling protocol such as the Constraint-based Routed Label Distribution Protocol (CR-LDP) or the extended Resource ReSerVation Protocol (RSVP-TE).

The remainder of this paper is organized as follows. The related works are introduced in section II. In section III the traffic bifurcation problem and the hop-count constrained traffic bifurcation problem with the node/link affinity condition are given. The results of the performance evaluation by simulation are discussed in section IV, and section V concludes this paper.

II. Related Work

In connection-oriented networks, [8] has analyzed the performance of multipath routing algorithms and has shown that the connection establishment time for reservation is significantly lowered in the multipath case. They didn't, however, fully consider the path computation problem. [9] has proposed a dynamic multipath routing algorithm in connection-oriented networks, where the shortest path is used under light traffic condition and multiple paths are utilized as the shortest path becomes congested. In this work, only connection or call-level, not flow-level routing and forwarding are considered. A Quality-of-Service (QoS) routing method via multiple paths under a time constraint is proposed when the bandwidth can be reserved, assuming all the reordered packets are recovered by optimal buffering at the receiver [10]. This scheme has much overhead for dynamic buffer adjustment at the receiver. The enhanced routing scheme for load balancing by separating long-lived and short-lived flows is proposed and it is shown that congestion can be greatly reduced [11]. It is shown that the quality of services can be enhanced by dividing the transport-level flows into UDP and TCP flows [12]. These works did not consider the path calculation problem. For the MPLS network, a traffic engineering method using multiple multipoint-to-point LSPs is proposed, which uses multiple routes as backup ones against failures [13]. Hence, the alternate paths are used only when primary routes do not work. The traffic bifurcation Linear Programming (LP) problem is formulated and heuristics for the non-bifurcating problem are proposed [14]. Although [14] minimizes the maximum of link utilization, it does not consider total network resources and constraints. Recently, Wang et al. have shown that the traffic bifurcation LP problem can be transformed into the shortest path problem by adjusting link weights in [15].

III. Problem Formulation

A. Traffic Bifurcation

The traffic bifurcation ($TB(g)$) problem consists of finding multiple paths carrying a part of or all the traffic between an ingress and an egress node which minimizes the maximum of link utilization, α.

When splitting a traffic demand to multiple paths, the granularity of load splitting, g ($0 < g \leq 1$) is defined to represent how coarsely a traffic demand can be divided. For example, when g is 0.1, splitting can be done in multiples of 0.1 (i.e., if we have three paths, splitting is done in 0.1, 0.3 and 0.6, respectively, for each path); while when g is 0.5, traffic bifurcation is done either for two links with equal load (0.5 each), or non-bifurcation occurs with one link.

The network is modeled as a directed graph, $G = (V, E)$, where V is the set of nodes and E represents the set of links. The capacity of a directional link $(i, j) \in E$ is c_{ij}. Each traffic demand ($k \in K$) is given for a node pair between an ingress router (s_k) and an egress router (t_k). The variable X^k_{ij} represents the fraction of the traffic demand k assigned to link (i, j). The integer variable M^k_{ij} represents how many units of basic discrete split demands for a traffic demand k are assigned to link (i, j). The maximum of M^k_{ij} will be $\lfloor \frac{1}{g} \rfloor$. Let d_k be a scaling factor to normalize total traffic demand from the source to become 1. The MIP problem for the bifurcation case is formulated as follows.

$$Minimize \quad \alpha$$

subject to

$$\sum_{j:(i,j)\in E} X^k_{ij} - \sum_{j:(i,j)\in E} X^k_{ji} = 0, k \in K, i \neq s_k, t_k \qquad (1)$$

$$\sum_{j:(i,j)\in E} X^k_{ij} - \sum_{j:(i,j)\in E} X^k_{ji} = 1, k \in K, i = s_k \qquad (2)$$

$$\sum_{j:(i,j)\in E} X^k_{ij} - \sum_{j:(i,j)\in E} X^k_{ji} = -1, k \in K, i = t_k \qquad (3)$$

$$\sum_{k\in K} d_k \cdot X^k_{ij} \leq c_{ij} \cdot \alpha, (i, j) \in E \qquad (4)$$

$$X^k_{ij} = M^k_{ij} \cdot g \qquad (5)$$

$$0 \leq X^k_{ij} \leq 1, 0 < g \leq 1, 0 \leq \alpha,$$
$$M^k_{ij} \in Z, 0 \leq M^k_{ij} \leq \lfloor 1/g \rfloor$$

The objective is to minimize α. Constraint (1), constraint (2) and constraint (3) represent the flow constraints for intermediate, source, and sink nodes, respectively. Constraint (4) is

the link capacity constraint. Constraint (5) states that assigned traffic demands are discrete.

The $TB(g)$ MIP problem can be solved by searching the branch-and-bound tree with an MIP solver such as CPLEX [18], and the solution gives the optimal flow values (X_{ij}^k). Especially, when g approaches to zero, the $TB(g)$ problem is reduced to an LP problem which can be solved with the classic Simplex method. Moreover, if g is 1, the $TB(g)$ problem is the integer programming problem for the non-bifurcating case.

The above $TB(g)$ problem finds the optimal $\bar{\alpha}$ and identifies the bottleneck link [14]. However, traffic in the network may still be reduced, because many flow assignment candidates satisfying $\bar{\alpha}$ exist. Moreover, cycles may occur, because the objective is to minimize not the total network resources but the maximum of link utilization. Therefore, the second step is performed after the above problem finds the optimal $\bar{\alpha}$. With a constant value of the maximum of link utilization, $\bar{\alpha}$, previously found by the $TB(g)$ formulation, we solve the multi-commodity problem with the following objective of minimizing the summation of flows assigned to each link under the fixed $\bar{\alpha}$, which removes the unnecessary traffic assignments and cycles.

$$Minimize \quad \sum_{(i,j)\in E} \sum_k X_{ij}^k$$

Finally, we use the shortest augmenting algorithm for the maximum flow problem [19] to derive multiple LSPs and splitting-load ratios of each traffic demand from fraction of traffic demand assigned to each link, $\{X_{ij}^k : X_{ij}^k > 0\}$.

B. Maximum Hop-count Constrained Traffic Bifurcation

The maximum hop-count constrained traffic bifurcation problem (denoted as $HTB(g, H)$) consists of finding hop-count constrained multiple paths between a source-destination pair with the objective of minimizing α. The granularity of the splitting load is represented by g, and H denotes the additional hop-count constraint compared to the shortest path[1]. The maximum hop-count allowed for LSPs of each traffic demand k is given as L_k[2].

Although the maximum hop-count constrained non-bifurcation problem can easily be formulated as the MIP problem by imposing $\sum_{(i,j)\in E} X_{ij}^k \le L_k, (X_{ij}^k = 0, 1)$ [16], a different problem formulation is necessary for the bifurcation case to simultaneously consider multiple paths. Therefore, we formulate the $HTB(g, H)$ MIP problem by using the hop-level flow conservation rule : the sum of incoming flows to a node i reached by the source with l hops equals to the sum of outgoing flows from the node i to adjacent nodes reached by the source with $(l+1)$ hops.

The same network model and the traffic demands are assumed as in the previous $TB(g)$ problem. Let X_{ij}^{kl} represent the fraction of a traffic demand k assigned on a link (i, j), where j is l hops far from s_k. The integer variable M_{ij}^{kl} represents how many discrete sub-loads of a traffic demand k are assigned to link (i, j) with l hops. The MIP formulation for the hop-count constrained traffic bifurcation problem is given as follows.

[1] In this paper, the shortest path means the minimum hop-count one.

[2] $L_k = L_{SH(k)} + H$, $L_{SH(k)}$ is the hop-count of the shortest path for each traffic demand k.

$$Minimize\ \alpha$$
subject to

$$\sum_{j:(i,j)\in E} X_{ij}^{kl} = \begin{cases} 1, & k \in K, i = s_k, l = 1 \\ 0, & k \in K, i \ne s_k, l = 1 \end{cases} \tag{6}$$

$$\sum_{j:(i,j)\in E} X_{ij}^{k(l+1)} - \sum_{j:(j,i)\in E} X_{ji}^{kl} = 0, \quad k \in K, i \ne s_k, t_k, 1 \le l < L_k \tag{7}$$

$$\sum_{l=1}^{L_k} \sum_{j:(j,i)\in E} X_{ji}^{kl} = 1, k \in K, i = t_k \tag{8}$$

$$\sum_{j:(i,j)\in E} X_{ij}^{kl} = 0, k \in K, i = t_k, \forall l \tag{9}$$

$$\sum_{l=1}^{L_k} \sum_{k\in K} d_k \cdot X_{ij}^{kl} \le c_{ij} \cdot \alpha, (i,j) \in E \tag{10}$$

$$X_{ij}^{kl} = M_{ij}^{kl} \cdot g \tag{11}$$

$$0 \le X_{ij}^{kl} \le 1, 0 \le g \le 1, 0 \le \alpha,$$
$$M_{ij}^{kl}, l \in Z, 0 \le M_{ij}^{kl} \le \lfloor 1/g \rfloor$$

The objective is to minimize α. Constraint (6) says that the total outgoing traffic over the first hop from the source is 1, and the total outgoing flow from intermediate nodes is 0. Constraint (7) is the hop-level flow constraint. Constraint (8) and (9) are the flow constraints for the destination node. Constraint (10) is the link capacity constraint. Constraint (11) is the discrete load assignment condition. In $HTB(g, H)$, if H is zero, traffic demand k will be assigned to the single shortest path or multiple equal-cost shortest paths. In the case of multiple shortest paths, traffic will be optimally bifurcated, whereas traditional ECMP routing will divide traffic evenly into multiple paths by 1/(the number of paths). In the same way as $TB(g)$, after the maximum of link utilization value ($\bar{\alpha}$) is found by the above MIP formulation, we solve the multi-commodity MIP problem with $\bar{\alpha}$ in order to minimize total network resources and to remove cycles. Finally, the same procedure to derive multiple LSPs and splitting-load ratios of each traffic demand as in $TB(g)$ is performed. Also, $HTB(g, H)$ will become either a continuous traffic bifurcation LP problem or a non-bifurcation integer programming problem according to the value of g.

In addition to the maximum number of the hop-count constraint, the node or link affinity constraint, such as including or excluding pre-defined nodes/links for a traffic demand, can be given according to administrative policy. The excluding node constraint (E_N^k) for a traffic demand k is formulated by using that the sum of outgoing and incoming flows to the node should be zero, and may be added to the $TB(g)$ and $HTB(g, H)$ problems.

To include specific nodes (I_N^k) for a traffic demand k, the outgoing or incoming flows to the included node should be greater than zero. Similarly, the link exclusion constraint (E_L^k) and inclusion constraint (I_{ij}^{kl}) for a traffic demand k may be added. In this paper, only the node exclusion constraint (E_N^k) of a traffic demand k is considered in order to avoid inefficient detouring.

IV. PERFORMANCE EVALUATION

A. Simulation Environment

The network topology shown in Fig. 1 represents the abstract US backbone topology [17]. Also, traffic demands given in [17] are used. In this network condition, the continuous traffic bifurcation LP problem and the discrete traffic bifurcation MIP problem are solved with CPLEX.

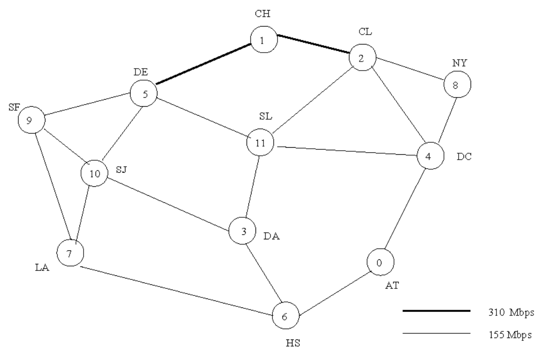


Fig. 1. Abstract US Network

B. Continuous Traffic Bifurcation

First, we investigated the performance of the continuous traffic bifurcation method by solving the LP problems. The different traffic engineering methods given below are compared to each other in terms of the maximum of link utilization (α), the total network resources ($R = \sum_{(i,j)\in E} X_{ij}^{k(l)}$), and the number of LSPs (P).

- **Shortest path based non-bifurcation** (SH): Each traffic demand is assigned along the shortest path from the ingress router to the egress one.
- **ECMP**: When multiple shortest paths exist, a traffic demand is evenly divided.
- **Traffic bifurcation** (TB): Without the hop-count constraint, each traffic demand is assigned by the LP solution of the continuous TB problem.
- **H hop constrained traffic bifurcation** ($HTB(H)$): All traffic demands are assigned by the LP solution of the $HTB(H)$ problem.
- **H hop constrained traffic bifurcation with node affinity** ($HTB{-}NA(H)$): The node affinity policy for a bifurcated traffic demand is added to $HTB(H)$.

In Table I, $HTB(1)$, which includes paths one hop longer than the shortest path, finds the same α of TB as TB, because many multiple paths are explored. $HTB(1)$ reduces α by 27.5 % over SH. Yet, $HTB(1)$ needs 15 additional LSPs and a 12.9 % increase in total network resources used. When the node exclusion constraint[3] is used, $HTB{-}NA(1)$ decreases α by 21.7 % with seven more LSPs, compared to SH. Nonetheless, the total network resources of $HTB{-}NA(1)$ are the same as those of SH. ECMP reduces α by 10.1 % compared to SH, but it requires 39.4 % more LSPs. It is interesting to note that $HTB(0)$ requires only 4.6 % more LSPs than SH, while decreasing α by 22.1 %.

From the simulation, it is shown that one additional hop-count constraint will be enough to find nearly the same α of TB for this network topology. The hop-count constraint should be selected for the specific network topology by policy. In addition, when hop-count constrained traffic bifurcation is combined with the appropriate node affinity policy, the maximum of link utilization can be greatly reduced with a few additional LSPs and network resources.

C. Discrete Traffic Bifurcation

When the granularity of dividing traffic is limited ($g \in \{0.1, 0.2, 0.25, 0.5, 1.0\}$), we solve the $TB(g)$ and $HTB(g, H)$ problems as MIP formulations and compare them with continuous traffic bifurcation (TB) and shortest path based traffic assignment (SH). The additional hop-count constraint (H) is set to one, because many multiple path candidates exist even with only one more hop.

The normalized $\hat{\alpha}_{TB}$ is defined to compare $TB(g)$ (or $HTB(g, H)$) with TB (or SH) ($\hat{\alpha}_{TB} = \frac{\alpha_{TB(g)} - \alpha_{TB}}{\alpha_{TB}}$). Similarly, the normalized total network resources, $\hat{R}_{SH}$ and the normalized number of LSPs, $\hat{P}_{SH}$ are calculated and compared to the shortest path based traffic allocation.

In Fig. 2-(a), it is shown that $TB(g)$ and $HTB(g, 1)$ find nearly the same α as TB regardless of g. This is because many alternate multiple paths for a traffic demand exist. Also, $TB(g)$ and $HTB(g, 1)$ reduce α by 26 % when compared to SH, as shown in Fig. 2-(a). As a useful measure of performance in practice, if we assume that each queue behaves as an $M/M/1$ queue of packets, the maximum queueing delay will be decreased to $\frac{1}{\mu(1-(\rho-0.26))}$, (where $\rho = \alpha$). Fig. 2-(b) indicates that the additional network resources required for multipath routing are not much, compared to the shortest path based traffic assignment scheme. With one additional hop-count constraint ($HTB(g, 1)$), the total network resources are increased by only 1.1 % in maximum when g is 0.25. In Fig. 2-(b), the number of LSPs necessary for bifurcation is compared to that of the shortest path based traffic assignment scheme. When g is 0.1, $TB(g)$ requires 21.2 % more LSPs than SH. However, $HTB(g, 1)$ needs 14.4 % more LSPs than SH in maximum when g is 0.25. Usually, as the granularity of splitting load becomes finer, more LSPs are used for minimizing α and the total network resources.

In order to examine the algorithm sensitivity to variations in traffic demand, we repeat the simulation with varied traffic demand. The upper bound of load variation ($v = \{10\%, 20\%, ..., \}$) is given, and the each entry is randomly altered within this range. For one value of v, ten sets of traffic demands are tested and averaged. In Fig. 2-(c), we examine the $HTB(g, H)$ algorithm sensitivity to variations in traffic demand. We see that $HTB(g, H)$ produces largely insensitive results to load variations.

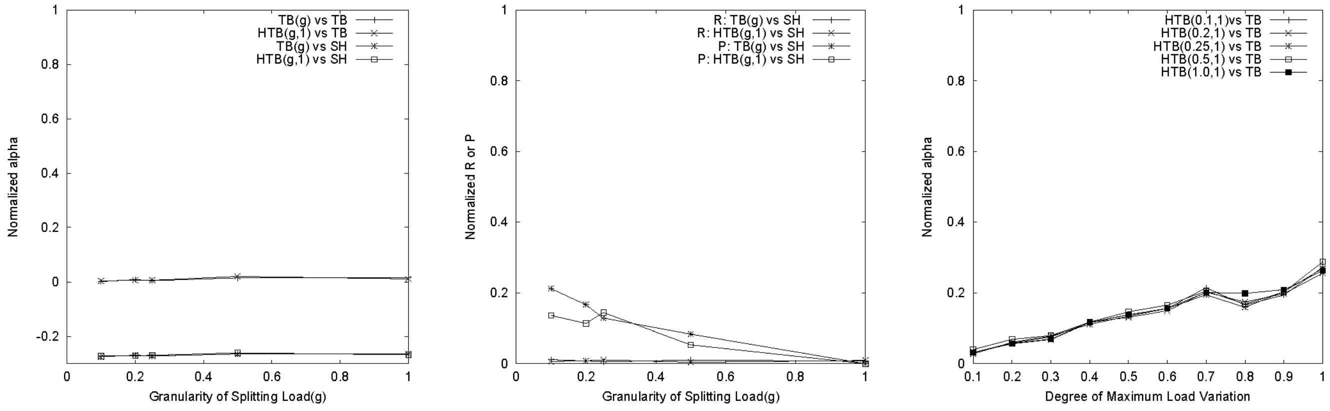
On the other hand, the hop-count constrained traffic bifurcation scheme greatly reduces the MIP solving time, because

[3] In the simulation, we excluded the bottleneck nodes after the node exclusion constraint is not applied to the problem ($E_N^{SF-CH} = \{LA, SJ\}$, $E_N^{SF-CL} = \{LA, SJ\}$, $E_N^{SF-DA} = \{LA\}$, $E_N^{SF-DC} = \{LA, SJ\}$, $E_N^{SF-DE} = \{SJ\}$, $E_N^{SF-NY} = \{LA, SJ\}$, $E_N^{SF-SJ} = \{LA\}$, $E_N^{SF-SL} = \{LA, SJ\}$).

TABLE I

MAXIMUM OF LINK UTILIZATION (α), TOTAL NETWORK RESOURCES (R), AND NUMBER OF LSPS (P) IN CONTINUOUS TRAFFIC BIFURCATION

	SH	ECMP	TB	HTB(0)	HTB(1)	HTB-NA(1)
α	1.22	1.10	0.88	0.95	0.88	0.95
R	4211.1	4211.1	4233	4211.1	4755.7	4211.1
P	132	218	153	138	147	139



(a) Normalized α (b) Normalized R and P (c) Sensitivity to load variation

Fig. 2. Normalized α, R, P, and load sensitivity in discrete traffic bifurcation

the branch-and-bound tree size is limited by the hop-count constraint. With CPLEX for the MIP solver[4], for example, $HTB(1.0, 1)$ takes only 8.8 seconds, whereas $TB(1.0)$ takes 1605.1 seconds.

Therefore, $HTB(g, H)$ is a practically useful traffic engineering scheme which finds multiple LSPs to minimize α without much increased network resources and LSPs. Sometimes, it is necessary that LSPs are reconfigured at minimal cost when the normalized α is greater than the threshold set by the administration policy.

V. CONCLUSION

In this paper, we propose multipath traffic engineering schemes for MPLS networks that minimize the maximum of link utilization, α. First, the traffic bifurcation problem formulated as an MIP problem ($TB(g)$) minimizes α by splitting a traffic demand to multiple LSPs. Although the proposed methods make splitting done at continuous or discrete levels, it is shown that the discretely splitting scheme which is suitable for easy implementation, with the granularity of splitting load, g, can get the near optimal solution of the continuously splitting scheme. Second, the hop-count constrained traffic bifurcation problem ($HTB(g, H)$) finds the LSPs which minimize α while satisfying the given hop-count constraint, H. The simulation results show that $HTB(g, H)$ solves nearly the same α of TB because many new candidate paths are derived. However, the network resources and LSPs required for $HTB(g, H)$ do not increase much, when compared to the shortest path based traffic assignment. Moreover, additional policy-based constraints such as including or excluding nodes/links can be supported in the proposed algorithm. The proposed traffic engineering scheme is practical and will be useful for reducing the probability of congestion by minimizing the utilization of the most heavily used link in the network.

[4]For the MIP solving parameter, the optimal gap is 0.01.

REFERENCES

[1] D. Bertsekas, and R. Gallager, Data Networks, Prentice Hall, 1992

[2] E. Rosen, A. Viswanathan, and R. Callon, "Multiprotocol Label Switching Architecture," Internet RFC3031, 2001

[3] D. Awduche, J. Malcolm, J. Agogbua, M. O'Dell, and J. McManus, "Requirements for Traffic Engineering Over MPLS," Internet RFC2702, 1999

[4] J. Moy, "OSPF Version 2," Internet RFC2328, 1998

[5] R. Callon, "Use of OSI IS-IS for Routing in TCP/IP and Dual Environments," Internet RFC1195, 1990

[6] D. Thaler and C. Hopps, "Multipath Issues in Unicast and Multicast Next-Hop Selection," Internet RFC2991, 2000

[7] Z. Cao, Z. Wang and E. Zegura, "Performance of Hashing-Based Schemes for Internet Load Balancing," INFOCOM'2000

[8] I. Cidon, R. Rom, and Y. Shavitt, "Analysis of Multi-Path Routing," IEEE/ACM Transactions on Networking, vol. 7, no. 6, pp. 885 - 896, Dec. 1999

[9] S. Bahk, and M. Zarki, "Dynamic Multi-path Routing and How it Compares with other Dynamic Routing Algorithms for High Speed Wide Area Networks," Computer Communications Review, vol. 22, no. 4, Oct. 1992

[10] N. S. V. Rao, and S. G. Batsell, "QoS Routing Via Multiple Paths Using Bandwidth Reservation," INFOCOM'98

[11] A. Shaikh, J. Rexford, and K. G. Shin, "Load-Sensitive Routing of Long-Lived IP Flows," SIGCOMM'99

[12] P. Bhaniramka, W. Sun, and R. Jain, "Quality of Service using Traffic Engineering over MPLS: An Analysis," LCN'2000

[13] H. Saito, Y. Miyao, and M. Yoshida, "Traffic Engineering using Multiple Multipoint-to-Point LSPs," INFOCOM'2000

[14] Y. Wang, and Z. Wang, "Explicit Routing Algorithms for Internet Traffic Engineering," ICCCN'99

[15] Z. Wang, Y. Wang, and L. Zhang, "Internet Traffic Engineering without Full Mesh Overlaying," INFOCOM'2001

[16] M. K. Girish, Z. Bei, and J. Hu, "Formulation of the Traffic Engineering Problems in MPLS based IP Networks," ISCC'2000

[17] Optimized Multipath, http://www.fictitious.org/omp

[18] CPLEX, http://www.cplex.com

[19] R. Ahuja, T. Magnanti, and J. Orlin, Network Flows, Prentice Hall, 1993.